UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/707,625	11/07/2000	Donald C. Englin	RA 5266	9920

7590 10/06/2003

Unisys Corporation
Charles A Johnson
P O Box 64942
MS 4773
St Paul, MN 55164

EXAMINER	
CHU, GABRIEL L	

ART UNIT	PAPER NUMBER
2184	

DATE MAILED: 10/06/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/707,625	Applicant(s) ENGLIN ET AL.	
	Examiner Gabriel L. Chu	Art Unit 2184	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *07 November 2000* .

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-22* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-22* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

11)☐ The proposed drawing correction filed on _____ is: a)☐ approved b)☐ disapproved by the Examiner.

If approved, corrected drawings are required in reply to this Office action.

12)☐ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

13)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____ .

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).

a) ☐ The translation of the foreign language provisional application has been received.

15)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____ .
4) ☐ Interview Summary (PTO-413) Paper No(s). _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: .

DETAILED ACTION

Claim Objections

1. Claims 5, 11, and 18 are objected to because of the following informalities:

Referring to line 2 of claim 5, the "stored tag address with the memory address access" is understood to refer to "the corrected tag address with the memory access address", correcting for antecedent errors.

Referring to the last lines of claims 11 and 18, Applicant has claimed "a current address", wherein a current address can refer to the "addresses currently cached", rendering "a current address" an unclear, if intended, antecedent. From the specification, Examiner understands that the second address compare module compares, instead, the "requested address".

Appropriate correction is required.

Claim Rejections - 35 USC § 102

2. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

3. Claims 1-7, 11, 12, 14, 15, 17-22 are rejected under 35 U.S.C. 102(b) as being anticipated by US 5509119 to La Fetra. Referring to claim 1, La Fetra discloses retrieving a stored tag address from the tag memory in response to the requester submitting a memory access address (From line 57 of column 3, " In operation, the CPU address Tag 201 is asserted by a CPU. The CPU-Tag is fed into the Tag input 207 of

the comparator and simultaneously fed into the index input 203 of the cache RAM 205. The index input is the address input of the cache RAM. The cache RAM then outputs the Cache-Tag and the ECC information 213 associated with the memory location in the cache RAM addressed by the index input. In this example, Cache-Tag 211 and its associated ECC 213 are presented on the outputs of the cache RAM and fed into the inputs 217 and 219 respectively of the Tag check and correct circuit 215."); performing a first comparison of the memory access address to the stored tag address to determine whether the requested data is stored in the cache memory (From line 2 of column 5, "A second comparator 401 is provided which receives the Cache-Tag / Cache-ECC pair from the cache RAM 205. Another input of the second comparator 401 receives the CPU address Tag 403 and a derived CPU-Tag ECC generated by an ECC generator 405. The ECC generator 405 takes the CPU-Tag and computes the appropriate ECC for that Tag in the same manner as is used to derive the Cache-ECC stored in the cache memory. This ECC generator derives the CPU-Tag ECC in the same or less time than is required to access the Cache-Tag/Cache-ECC pair from the cache RAM. The second comparator 401 compares the Cache-Tag to the CPU-Tag and compares the derived CPU-Tag ECC to the Cache-ECC. If both the Tags and the ECCs match, then the comparator 401 outputs a fast hit signal 407. Since the ECC generator derives the CPU-Tag ECC in parallel with the cache RAM access, the time required to assert the fast hit signal is less than required to assert the true hit signal by the delay time through the Tag check and correction circuit 215 (approximately 2 ns)."); monitoring for errors in the stored tag address, wherein the monitoring is performed

contemporaneously with the first comparison of the memory access address and the stored tag address (From line 1 of column 4, "The Tag check and correct circuit 215 checks the Cache-Tag for errors using the ECC information and corrects the Cache-Tag if an error is found. This checked and corrected Tag is then presented on the output 221 of the circuit and fed to the corrected Tag input 223 of the Tag comparator 209. After the corrected Tag is available to the Tag comparator 209, the comparator compares the corrected Tag to the CPU-Tag and if the two Tags match, the comparator outputs a true hit signal on output line 225. This signal is used by the cache memory system to provide data in the cache memory to the CPU by methods known in the art. If the two Tags do not match, then there has been a cache miss and the cache is updated as previously discussed."); if a tag address error is detected, disregarding the first comparison, correcting the tag address error, and performing a second comparison of the memory access address and the corrected tag address to determine whether the requested data is stored in the cache memory (From line 25 column 5, "If the fast hit is not asserted, then either there is a cache miss or the Cache-Tag has an error and the cache memory system waits for the true hit/miss signal as before."); and if no tag address error is detected, utilizing results of the first comparison to determine whether the requested data is stored in the cache memory (From line 22 column 5, "Since, most of the time the cache tag does not need to be corrected and generally there is a cache hit, by using the fast hit signal, the cache memory system can supply data to the CPU in less time than the prior art designs allow.").

Referring to claim 2, La Fetra discloses monitoring for errors comprises storing a

single error correction code with the data stored in the tag memory (From figure 2, element 205.).

Referring to claim 3, La Fetra discloses the single error correction code is coded to provide error detection for the stored tag address and a plurality of configuration fields (From line 40 of column 3, "FIG. 1 illustrates a typical cache Tag-ECC entry 101 having three primary segments 103, 105 and 107. Segment 105 contains "house keeping" information such as bits that indicate the cache is "dirty" or "private". The ECC field 107 has error correcting data for both the Tag segment 103 and the house keeping segment 105.").

Referring to claim 4, La Fetra discloses the first comparison compares only the stored tag address with the memory access address, and disregards comparison of any stored error correction code bits (From line 2 of column 5, "A second comparator 401 is provided which receives the Cache-Tag / Cache-ECC pair from the cache RAM 205. Another input of the second comparator 401 receives the CPU address Tag 403 and a derived CPU-Tag ECC generated by an ECC generator 405. The ECC generator 405 takes the CPU-Tag and computes the appropriate ECC for that Tag in the same manner as is used to derive the Cache-ECC stored in the cache memory. This ECC generator derives the CPU-Tag ECC in the same or less time than is required to access the Cache-Tag/Cache-ECC pair from the cache RAM. The second comparator 401 compares the Cache-Tag to the CPU-Tag and compares the derived CPU-Tag ECC to the Cache-ECC. If both the Tags and the ECCs match, then the comparator 401 outputs a fast hit signal 407. Since the ECC generator derives the CPU-Tag ECC in

parallel with the cache RAM access, the time required to assert the fast hit signal is less than required to assert the true hit signal by the delay time through the Tag check and correction circuit 215 (approximately 2 ns)." Wherein the second comparator 401 disclosed by La Fetra is comprised of two comparisons, the Cache-Tag/CPU-Tag comparison and the Cache-ECC/CPU-ECC comparison. Further, from line 32 of column 4, "Given an address asserted by the CPU (CPU-Tag), the information that must be in the cache entry 101 for a cache hit to occur is predictable. That is, the Cache-Tag must match the CPU-Tag.").

Referring to claim 5, La Fetra discloses the second comparison compares only the requested tag address with the memory access address, and disregards comparison of any stored error correction code bits (From line 1 of column 4, "The Tag check and correct circuit 215 checks the Cache-Tag for errors using the ECC information and corrects the Cache-Tag if an error is found. This checked and corrected Tag is then presented on the output 221 of the circuit and fed to the corrected Tag input 223 of the Tag comparator 209. After the corrected Tag is available to the Tag comparator 209, the comparator compares the corrected Tag to the CPU-Tag and if the two Tags match, the comparator outputs a true hit signal on output line 225.").

Referring to claim 6, La Fetra discloses performing the first comparison and monitoring for errors in the stored tag address occur contemporaneously with correcting the tag address error and performing the second comparison (From figure 4, the Fast Hit 407 and True Hit 225 are shown to operate in parallel. Further, from line 16 of column 25, "Since the ECC generator derives the CPU-Tag ECC in parallel with the

cache RAM access, the time required to assert the fast hit signal is less than required to assert the true hit signal by the delay time through the Tag check and correction circuit 215 (approximately 2 ns). Since, most of the time the cache tag does not need to be corrected and generally there is a cache hit, by using the fast hit signal, the cache memory system can supply data to the CPU in less time than the prior art designs allow. If the fast hit is not asserted, then either there is a cache miss or the Cache-Tag has an error and the cache memory system waits for the true hit/miss signal as before.").

Referring to claim 7, La Fetra discloses correcting the tag address error and performing the second comparison are initiated upon recognition of a tag address error (From line 1 of column 4, "The Tag check and correct circuit 215 checks the Cache-Tag for errors using the ECC information and corrects the Cache-Tag if an error is found. This checked and corrected Tag is then presented on the output 221 of the circuit and fed to the corrected Tag input 223 of the Tag comparator 209. After the corrected Tag is available to the Tag comparator 209, the comparator compares the corrected Tag to the CPU-Tag and if the two Tags match, the comparator outputs a true hit signal on output line 225.").

Referring to claim 11, La Fetra discloses a cache hit detector, comprising: (a) a tag memory to store tag addresses corresponding to addresses currently cached (From figure 4, element 205); (b) a fast hit detection circuit, comprising: (i) a first address compare module coupled to the tag memory to receive a tag address and to compare the tag address to a requested address (From line 2 of column 5, "A second comparator 401 is provided which receives the Cache-Tag / Cache-ECC pair from the cache RAM

205. Another input of the second comparator 401 receives the CPU address Tag 403 and a derived CPU-Tag ECC generated by an ECC generator 405. The ECC generator 405 takes the CPU-Tag and computes the appropriate ECC for that Tag in the same manner as is used to derive the Cache-ECC stored in the cache memory. This ECC generator derives the CPU-Tag ECC in the same or less time than is required to access the Cache-Tag/Cache-ECC pair from the cache RAM. The second comparator 401 compares the Cache-Tag to the CPU-Tag and compares the derived CPU-Tag ECC to the Cache-ECC. If both the Tags and the ECCs match, then the comparator 401 outputs a fast hit signal 407. Since the ECC generator derives the CPU-Tag ECC in parallel with the cache RAM access, the time required to assert the fast hit signal is less than required to assert the true hit signal by the delay time through the Tag check and correction circuit 215 (approximately 2 ns)."); (ii) an error detector coupled to the tag memory to receive the tag address and to determine whether there are any errors in the tag address (From line 2 of column 5, "A second comparator 401 is provided which receives the Cache-Tag / Cache-ECC pair from the cache RAM 205. Another input of the second comparator 401 receives the CPU address Tag 403 and a derived CPU-Tag ECC generated by an ECC generator 405. The ECC generator 405 takes the CPU-Tag and computes the appropriate ECC for that Tag in the same manner as is used to derive the Cache-ECC stored in the cache memory. This ECC generator derives the CPU-Tag ECC in the same or less time than is required to access the Cache-Tag/Cache-ECC pair from the cache RAM. The second comparator 401 compares the Cache-Tag to the CPU-Tag and compares the derived CPU-Tag ECC to the Cache-

ECC. If both the Tags and the ECCs match, then the comparator 401 outputs a fast hit signal 407. Since the ECC generator derives the CPU-Tag ECC in parallel with the cache RAM access, the time required to assert the fast hit signal is less than required to assert the true hit signal by the delay time through the Tag check and correction circuit 215 (approximately 2 ns)."); (iii) a gated output module coupled to the first address compare module and the error detector to output a fast hit indication if and only if no error is discovered by the error detector and the requested address is stored in the tag memory (From line 2 of column 5, "A second comparator 401 is provided which receives the Cache-Tag / Cache-ECC pair from the cache RAM 205. Another input of the second comparator 401 receives the CPU address Tag 403 and a derived CPU-Tag ECC generated by an ECC generator 405. The ECC generator 405 takes the CPU-Tag and computes the appropriate ECC for that Tag in the same manner as is used to derive the Cache-ECC stored in the cache memory. This ECC generator derives the CPU-Tag ECC in the same or less time than is required to access the Cache-Tag/Cache-ECC pair from the cache RAM. The second comparator 401 compares the Cache-Tag to the CPU-Tag and compares the derived CPU-Tag ECC to the Cache-ECC. If both the Tags and the ECCs match, then the comparator 401 outputs a fast hit signal 407." Further, line 25 of column, "If the fast hit is not asserted, then either there is a cache miss or the Cache-Tag has an error and the cache memory system waits for the true hit/miss signal as before."); (c) a slow hit detection circuit, comprising: (i) an error correction circuit coupled to the tag memory to receive the tag address and to correct errors in the tag address (From figure 4, element 215.); and (ii) a second

address compare module coupled to the error correction circuit to receive the corrected tag address and to compare the corrected tag address to the requested address (From figure 4, element 209.).

Referring to claim 12, La Fetra discloses the fast hit detection circuit and the slow hit detection circuit are coupled in parallel such that the first address compare module and the error detector perform operations concurrently with operations of the second address compare module and the error correction circuit (From figure 4, the Fast Hit 407 and True Hit 225 are shown to operate in parallel. Further, from line 16 of column 25, "Since the ECC generator derives the CPU-Tag ECC in parallel with the cache RAM access, the time required to assert the fast hit signal is less than required to assert the true hit signal by the delay time through the Tag check and correction circuit 215 (approximately 2 ns). Since, most of the time the cache tag does not need to be corrected and generally there is a cache hit, by using the fast hit signal, the cache memory system can supply data to the CPU in less time than the prior art designs allow. If the fast hit is not asserted, then either there is a cache miss or the Cache-Tag has an error and the cache memory system waits for the true hit/miss signal as before.").

Referring to claim 14, La Fetra discloses the tag memory further stores an error correction code for each block of data stored in the tag memory, wherein each block of data is associated with a single error correction code, and the single error correction code provides error correction capabilities for the stored tag address and a plurality of configuration fields (From line 40 of column 3, "FIG. 1 illustrates a typical cache Tag-ECC entry 101 having three primary segments 103, 105 and 107. Segment 105

contains "house keeping" information such as bits that indicate the cache is "dirty" or "private". The ECC field 107 has error correcting data for both the Tag segment 103 and the house keeping segment 105.").

Referring to claim 15, La Fetra discloses the first address compare module and the error detector are coupled in parallel to contemporaneously compare the tag address to a requested address and determine whether there are any errors in the tag address (From line 2 of column 5, "A second comparator 401 is provided which receives the Cache-Tag / Cache-ECC pair from the cache RAM 205. Another input of the second comparator 401 receives the CPU address Tag 403 and a derived CPU-Tag ECC generated by an ECC generator 405. The ECC generator 405 takes the CPU-Tag and computes the appropriate ECC for that Tag in the same manner as is used to derive the Cache-ECC stored in the cache memory. This ECC generator derives the CPU-Tag ECC in the same or less time than is required to access the Cache-Tag/Cache-ECC pair from the cache RAM. The second comparator 401 compares the Cache-Tag to the CPU-Tag and compares the derived CPU-Tag ECC to the Cache-ECC. If both the Tags and the ECCs match, then the comparator 401 outputs a fast hit signal 407. Since the ECC generator derives the CPU-Tag ECC in parallel with the cache RAM access, the time required to assert the fast hit signal is less than required to assert the true hit signal by the delay time through the Tag check and correction circuit 215 (approximately 2 ns)." Wherein the second comparator 401 disclosed by La Fetra is comprised of two comparisons, the Cache-Tag/CPU-Tag comparison and the Cache-ECC/CPU-ECC comparison.).

Referring to claim 17, La Fetra discloses the error detector determines whether there are any single bit errors in the tag address (From line 2 of column 5, "A second comparator 401 is provided which receives the Cache-Tag / Cache-ECC pair from the cache RAM 205. Another input of the second comparator 401 receives the CPU address Tag 403 and a derived CPU-Tag ECC generated by an ECC generator 405. The ECC generator 405 takes the CPU-Tag and computes the appropriate ECC for that Tag in the same manner as is used to derive the Cache-ECC stored in the cache memory. This ECC generator derives the CPU-Tag ECC in the same or less time than is required to access the Cache-Tag/Cache-ECC pair from the cache RAM. The second comparator 401 compares the Cache-Tag to the CPU-Tag and compares the derived CPU-Tag ECC to the Cache-ECC. If both the Tags and the ECCs match, then the comparator 401 outputs a fast hit signal 407. Since the ECC generator derives the CPU-Tag ECC in parallel with the cache RAM access, the time required to assert the fast hit signal is less than required to assert the true hit signal by the delay time through the Tag check and correction circuit 215 (approximately 2 ns)." Wherein any comparison that is not equal constitutes at least a single bit error.).

Referring to claim 18, La Fetra discloses a data processing system comprising: (a) a main memory module for storing data (From line 14 of column 1, "a main memory".); (b) at least one cache memory coupled to the main memory module to cache at least a portion of the data stored in the main memory module (From line 20 of column 1, "a cache".); (c) at least one processing unit (From line 13 of column 1, "a central processing unit".) to process data and to control data access with the main

memory module and the cache memory, the processing unit comprising: (1) a tag memory to store tag addresses corresponding to addresses currently cached (From figure 4, element 205.); (2) a fast hit detection circuit, comprising: (i) a first address compare module coupled to the tag memory to receive a tag address and to compare the tag address to a requested address (From line 2 of column 5, "A second comparator 401 is provided which receives the Cache-Tag / Cache-ECC pair from the cache RAM 205. Another input of the second comparator 401 receives the CPU address Tag 403 and a derived CPU-Tag ECC generated by an ECC generator 405. The ECC generator 405 takes the CPU-Tag and computes the appropriate ECC for that Tag in the same manner as is used to derive the Cache-ECC stored in the cache memory. This ECC generator derives the CPU-Tag ECC in the same or less time than is required to access the Cache-Tag/Cache-ECC pair from the cache RAM. The second comparator 401 compares the Cache-Tag to the CPU-Tag and compares the derived CPU-Tag ECC to the Cache-ECC. If both the Tags and the ECCs match, then the comparator 401 outputs a fast hit signal 407. Since the ECC generator derives the CPU-Tag ECC in parallel with the cache RAM access, the time required to assert the fast hit signal is less than required to assert the true hit signal by the delay time through the Tag check and correction circuit 215 (approximately 2 ns)."); (ii) an error detector coupled to the tag memory to receive the tag address and to determine whether there are any errors in the tag address (From line 2 of column 5, "A second comparator 401 is provided which receives the Cache-Tag / Cache-ECC pair from the cache RAM 205. Another input of the second comparator 401 receives the CPU address Tag 403 and a derived CPU-Tag

ECC generated by an ECC generator 405. The ECC generator 405 takes the CPU-Tag and computes the appropriate ECC for that Tag in the same manner as is used to derive the Cache-ECC stored in the cache memory. This ECC generator derives the CPU-Tag ECC in the same or less time than is required to access the Cache-Tag/Cache-ECC pair from the cache RAM. The second comparator 401 compares the Cache-Tag to the CPU-Tag and compares the derived CPU-Tag ECC to the Cache-ECC. If both the Tags and the ECCs match, then the comparator 401 outputs a fast hit signal 407. Since the ECC generator derives the CPU-Tag ECC in parallel with the cache RAM access, the time required to assert the fast hit signal is less than required to assert the true hit signal by the delay time through the Tag check and correction circuit 215 (approximately 2 ns)."); (iii) a gated output module coupled to the first address compare module and the error detector to output a fast hit indication if and only if no error is discovered by the error detector and the requested address is stored in the tag memory (From line 2 of column 5, "A second comparator 401 is provided which receives the Cache-Tag / Cache-ECC pair from the cache RAM 205. Another input of the second comparator 401 receives the CPU address Tag 403 and a derived CPU-Tag ECC generated by an ECC generator 405. The ECC generator 405 takes the CPU-Tag and computes the appropriate ECC for that Tag in the same manner as is used to derive the Cache-ECC stored in the cache memory. This ECC generator derives the CPU-Tag ECC in the same or less time than is required to access the Cache-Tag/Cache-ECC pair from the cache RAM. The second comparator 401 compares the Cache-Tag to the CPU-Tag and compares the derived CPU-Tag ECC to the Cache-

ECC. If both the Tags and the ECCs match, then the comparator 401 outputs a fast hit signal 407." Further, line 25 of column, "If the fast hit is not asserted, then either there is a cache miss or the Cache-Tag has an error and the cache memory system waits for the true hit/miss signal as before."); (3) a slow hit detection circuit, comprising: (i) an error correction circuit coupled to the tag memory to receive the tag address and to correct errors in the tag address (From figure 4, element 215.); and (ii) a second address compare module coupled to the error correction circuit to receive the corrected tag address and to compare the corrected tag address to the requested address (From figure 4, element 209.).

Referring to claim 19, La Fetra discloses the fast hit detection circuit and the slow hit detection circuit are configured in parallel such that the first address compare module and the error detector perform operations concurrently with operations of the second address compare module and the error correction circuit (From figure 4, the Fast Hit 407 and True Hit 225 are shown to operate in parallel. Further, from line 16 of column 25, "Since the ECC generator derives the CPU-Tag ECC in parallel with the cache RAM access, the time required to assert the fast hit signal is less than required to assert the true hit signal by the delay time through the Tag check and correction circuit 215 (approximately 2 ns). Since, most of the time the cache tag does not need to be corrected and generally there is a cache hit, by using the fast hit signal, the cache memory system can supply data to the CPU in less time than the prior art designs allow. If the fast hit is not asserted, then either there is a cache miss or the Cache-Tag has an error and the cache memory system waits for the true hit/miss signal as before.").

Referring to claim 20, La Fetra discloses the tag memory further stores an error correction code associated with each block of data, wherein each block of data is associated with a single error correction code, and the single error correction code! provides error correction capabilities for the stored tag address and a plurality of configuration fields (From line 40 of column 3, "FIG. 1 illustrates a typical cache Tag-ECC entry 101 having three primary segments 103, 105 and 107. Segment 105 contains "house keeping" information such as bits that indicate the cache is "dirty" or "private". The ECC field 107 has error correcting data for both the Tag segment 103 and the house keeping segment 105.").

Referring to claim 21, La Fetra discloses a cache hit detector, comprising: (a) means for storing tag memory addresses corresponding to addresses of data currently stored in cache memory (From figure 4, element 205.); (b) means for providing alternate cache hit detection via concurrent processing on each of at least two cache hit detection paths, the alternate cache hit detection means comprising: (1) first hit detection path means for detecting cache hits without first performing error detection and correction (From figure 4, "Fast Hit" path 407.); (2) second hit detection path means for detecting cache hits, the second hit detection path means comprising: (i) means for detecting errors in the tag memory address (From figure 4, element 215.); (ii) means for correcting the tag memory address if errors in the tag memory address are discovered (From figure 4, element 215.); (iii) means for detecting for cache hits using the corrected tag memory address if errors in the tag memory address are discovered (From figure 4, element 209.); and (iv) means for disabling the first hit detection path means if errors in

the tag memory address are discovered (From line 25 column 5, "If the fast hit is not asserted, then either there is a cache miss or the Cache-Tag has an error and the cache memory system waits for the true hit/miss signal as before.").

Referring to claim 22, La Fetra discloses means for coordinating timing between the first hit detection path means and the second hit detection path means (From line 22 of column 5, "Since, most of the time the cache tag does not need to be corrected and generally there is a cache hit, by using the fast hit signal, the cache memory system can supply data to the CPU in less time than the prior art designs allow. If the fast hit is not asserted, then either there is a cache miss or the Cache-Tag has an error and the cache memory system waits for the true hit/miss signal as before.").

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 8-10, 13, and 16 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 5509119 to La Fetra as applied to claims 1 and 11 above. Referring to claims 8 and 9, La Fetra discloses disregarding the first comparison results of the first comparison through an output gate (From line 25 column 5, "If the fast hit is not asserted, then either there is a cache miss or the Cache-Tag has an error and the cache memory system waits for the true hit/miss signal as before."). Although La Fetra does not specifically disclose said disregarding comprises blocking passage of the

results by providing an error signal to the result signal's output gate when a tag address error is detected and disabling an output of the output gate upon receipt of the error signal, blocking a signal on error is notoriously well known in the art. A person of ordinary skill in the art at the time of the invention would have been motivated to block a signal on error because, from line 25 of column 5, "If the fast hit is not asserted, then either there is a cache miss or the Cache-Tag has an error and the cache memory system waits for the true hit/miss signal as before."

Referring to claim 10, La Fetra discloses enabling the tag address error to be corrected and the second comparison of the memory access and corrected tag addresses to be performed in response to the error signal (From line 22 column 5, "Since, most of the time the cache tag does not need to be corrected and generally there is a cache hit, by using the fast hit signal, the cache memory system can supply data to the CPU in less time than the prior art designs allow.").

Referring to claim 13, La Fetra discloses means to coordinate timing between the fast hit detection circuit and the slow hit detection circuit (From line 25 column 5, "If the fast hit is not asserted, then either there is a cache miss or the Cache-Tag has an error and the cache memory system waits for the true hit/miss signal as before."). Although La Fetra does not specifically disclose latching as this means, storing for subsequent comparison is notoriously well known in the art. A person of ordinary skill in the art at the time of the invention would have been motivated to store a result for subsequent comparison because it cannot or should not be used when it is immediately available.

Referring to claim 16, La Fetra discloses first address comparison results and a

resulting error indicator signal (From line 2 of column 5, "A second comparator 401 is provided which receives the Cache-Tag / Cache-ECC pair from the cache RAM 205. Another input of the second comparator 401 receives the CPU address Tag 403 and a derived CPU-Tag ECC generated by an ECC generator 405. The ECC generator 405 takes the CPU-Tag and computes the appropriate ECC for that Tag in the same manner as is used to derive the Cache-ECC stored in the cache memory. This ECC generator derives the CPU-Tag ECC in the same or less time than is required to access the Cache-Tag/Cache-ECC pair from the cache RAM. The second comparator 401 compares the Cache-Tag to the CPU-Tag and compares the derived CPU-Tag ECC to the Cache-ECC."). Although La Fetra does not specifically disclose latching the comparison and error indicator signal, wherein the comparison results and the error indicator signal are not passed to the gated output until both the comparison results and the error indicator signal are available at the latches, and until simultaneously clocked to concurrently provide the comparison results and the error indicator signal to the gated output, storing for subsequent comparison and gating or clocking a signal are notoriously well known in the art. A person of ordinary skill in the art at the time of the invention would have been motivated to store for subsequent comparison and to gate a signal because it cannot or should not be used when it is immediately available and to allow a signal to settle, or reach steady-state.

Conclusion

6. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

US 4483003 to Beal

US 5353424 to Partovi et al.

US 5724528 to Kulik et al.

US 5926484 to Takusagawa

US 6038693 to Zhang

US 6226763 to Fu et al.

US 6477635 to Kahle et al.

US 6546501 to Hataida et al.

JP 01273152 A to Ito et al.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Gabriel L. Chu whose telephone number is (703) 308-7298. The examiner can normally be reached on weekdays with alternate Fridays off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Robert W. Beausoliel, Jr. can be reached on (703) 305-9713. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

gc

ROBERT BEAUSOLIEL
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100